AMENDMENTS TO THE SPECIFICATION

Please amend paragraph 0001 as follows:

[0001] This Application is related to the following two co-pending United States Patent Applications: U.S. Patent Application No. 10/766,116, entitled "Charge Packet Metering For Coarse/Fine Programming Of Non-Volatile Memory," by Daniel C. Guterman, Nima Mokhlesi and Yupin Fong, filed January 27, 2004 ~~the same day as the present application~~; and U.S. Patent Application No. 10/766,217, entitled "Efficient Verification For Coarse/Fine Programming Of Non-Volatile Memory," by Daniel C. Guterman, Nima Mokhlesi and Yupin Fong, filed January 27, 2004 ~~the same day as the present application~~. The two above-listed related applications are both incorporated herein by reference in their entirety.

Please amend paragraph 0006 as follows:

[0006] Typically, the program voltage applied to the control gate is applied as a series of pulses. The magnitude of the pulses is increased with each successive pulse by a predetermined step size (e.g. 0.2v). In the periods between the pulses, verify operations are carried out. That is, the programming level of each cell of a group of cells being programmed in parallel is read between successive programming pulses to determine whether it is equal to or greater than a verify level to which it is being programmed. One means of verifying the programming is to test conduction at a specific compare point. The cells that are verified to be sufficiently programmed are locked out, for example in NAND cells, by raising the bit line voltage from 0 to Vdd (e.g., 2.5 volts) to stop the programming process for those cells. In some cases, the number of pulses will be limited (e.g. 20 pulses) and if a given memory cell is not completely programmed by the last pulse, then an error is assumed. In some implementations, memory cells are erased (in blocks or other units) prior to programming. More information about programming can be found in U.S. Patent Application 10/379,608, titled "Self Boosting Technique," filed on March 5, 2003, issued as U.S. Patent No. 6,859,397 on February 22, 2005; and in U.S. Patent Application 10/629,068, titled "Detecting Over Programmed Memory," filed on July 29, 2003, published as Publication No. US2005/0024939 on February 3, 2005, both of which ~~applications~~ are incorporated herein by reference in their entirety.

Please amend paragraph 0008 as follows:

[0008] A multi-state flash memory cell is implemented by identifying multiple, distinct allowed threshold voltage ranges separated by forbidden voltage ranges. For example, Figure 2 shows eight threshold ranges (0, 1, 2, 3, 4, 5, 6, 7), corresponding to three bits of data. Other memory cells can use more than eight threshold ranges or less than eight threshold ranges. Each distinct threshold voltage range corresponds to predetermined values for the set of data bits. In some implementations, these data values (e.g. logical states) are assigned to the threshold ranges using a gray code assignment so that if the threshold voltage of a floating gate erroneously shifts to its neighboring physical state, only one bit will be affected. The specific relationship between the data programmed into the memory cell and the threshold voltage ranges of the cell depends upon the data encoding scheme adopted for the cells. For example, U.S. Patent No. 6,222,762 and U.S. Patent Application No. 10/461,244, "Tracking Cells For A Memory System," filed on June 13, 2003, published as Publication No. US2004/0255090 on December 16, 2004, both of which are incorporated herein by reference in their entirety, describe various data encoding schemes for multi-state flash memory cells.

Please amend paragraph 0010 as follows:

[0009] Performing seven verify operations after each programming pulses slows down the programming process. One means for reducing the time burden of verifying is to use a more efficient verify process. For example, in U.S. Patent Application Serial No. 10/314,055, "Smart Verify for Multi-State Memories," filed December 5, 2002, published as Publication No. US2004/0109362 on June 10, 2004, incorporated herein by reference in its entirety, a Smart Verify process is disclosed. In an exemplary embodiment of the write sequence for the multi-state memory during a program/verify sequence using the Smart Verify process, at the beginning of the process only the lowest state (e.g. state 1 of Fig. 2) of the multi-state range to which the selected memory cells are being programmed is checked during the verify phase. Once the first storage state (e.g. state 1 of Fig. 2) is reached by one or more of the memory cells, the next state (e.g. state 2 of Fig. 2) in the sequence of multi-states is added to the verify process. This next state can either be added immediately upon the fastest cells reaching this preceding state in the sequence or, since memories are generally designed to have several programming steps to move from state to state, after a delay of several cycles. The amount of delay can either be fixed or use a parameter based implementation, which allows the amount of delay to be set according to device characteristics. The adding of states to the set being checked in the verify phase continues as per above until the highest state has been added. Similarly, lower states can be removed from the verify set as all of the memory cells bound for these levels verify successfully to those target values and are locked out from further programming.

Please amend paragraph 0012 as follows:

[0012] One solution for achieving tight threshold distributions without unreasonably slowing down the programming process is to use a two phase programming process. The first phase, a coarse programming phase, includes attempts to raise the threshold voltage in a faster manner and paying relatively less attention to achieving a tight threshold distribution. The second phase, a fine programming phase, attempts to raise the threshold voltage in a slower manner in order to reach the target threshold voltage while also achieving a tighter threshold distribution. Example of coarse/fine programming methodologies can be found in the following patent documents that are incorporated herein by reference in their entirety: U.S. Patent Application No. 10/051,372, "Non-Volatile Semiconductor Memory Device Adapted to Store A Multi-Valued Data in a Single Memory Cell," filed January 22, 2002, issued as U.S. Patent No. 6,643,188 on November 4, 2003; U.S. Patent 6,301,161; U.S. Patent 5,712,815; U.S. Patent No. 5,220,531; and U.S. Patent No. 5,761,222. When verifying a memory cell during programming, some prior solutions will first perform the verify process for the coarse mode and then subsequently perform the verify process for the fine mode. Such a verification process increases the time needed for verification. The coarse/fine programming methodology can be used in conjunction with the Smart Verify process described above.